Data Mining

Data Lake

Claudio Sartori

DISI
Department of Computer Science and Engineering – University of Bologna, Italy
claudio.sartori@unibo.it

New demands of the digital era

- Granular personalization, individual treatment
- Recommendations
- Instant, real–time processing and decision making
- 360 degree view of a customer
- Contextual insight
- Actionable insight
- Smart, analytics–enabled services
- Ability to adapt to changes in the business landscape

New data types and dark data

Source: Eagle Alpha

Dark Data

- Acquired and stored through computer–based operations
- Not used in the decision–making process
- Estimates:
  - 90% of sensor and analog–to–digital conversions never used
  - In some companies 99% of data never analysed
  - "The storage environments of EMEA organizations consist of 54% dark data, 32% redundant, obsolete and trivial data and 14% business-critical data. By 2020, this can add up to $891 billion in storage and management costs that can otherwise be avoided."

sources: Wikipedia and Datamation

What is a Data Lake

Source for the following slides: Utkin and Komissarov, DataArt

- A repository of data stored in its natural/raw format
  - usually *object blobs* or *files*
- Dive anywhere, flexible access, schema on read
  - diverse toolsets and processing types based on data and use case
- Quickly ingest anything, do not enforce schema on write
  - Big Data ingestion
- All data in one place
  - a single source of truth for source data
- Low cost scalable storage
  - decoupled from computing facilities
- Future proof

New data $\Rightarrow$

new business opportunity

- Massive personalisation of products, services and market channels. B2C in any industry
- Real time and Productive risk analytics
- Operation optimisation
  - predictive maintenance
  - supply chain optimisation

Traditional data analytics architecture

- Siloed departmental data
- Enterprise Data Warehouse, long implementation cycles
- Limits of scalability and high costs
  - Compute and storage have to be scaled together
- Lack of support for
  - unstructured data
  - ML/DS workloads
  - ad-hoc analytical queries

ODS = Operational Data Store

Requirements

- Wide range of analytics use cases
- Data in hands of the business users
- Flexible and scalable data architecture
- 20% structured DW/BI workloads / 80% unstructured data, ad–hoc and data science workloads
- Short time to value
- Holistic metadata management, governance, security monitoring and usage analytics

Features and use cases of the technologies

Use cases	Tech solutions	Feature trends
Mission critical, low latency, insight apps	Data Warehouse / Hot	More expensive HW/SW Use cases specific data Less latency More governance Higher data quality Used by end-users and data analysts
Agile insight apps	Data Hub / Warm	↑↓ Less expensive HW/SW All enterprise data More latency Less governance Lower data quality Used by data scientists
Staging area, data mining, searching, profiling, cataloging	Data Lake / Cold	

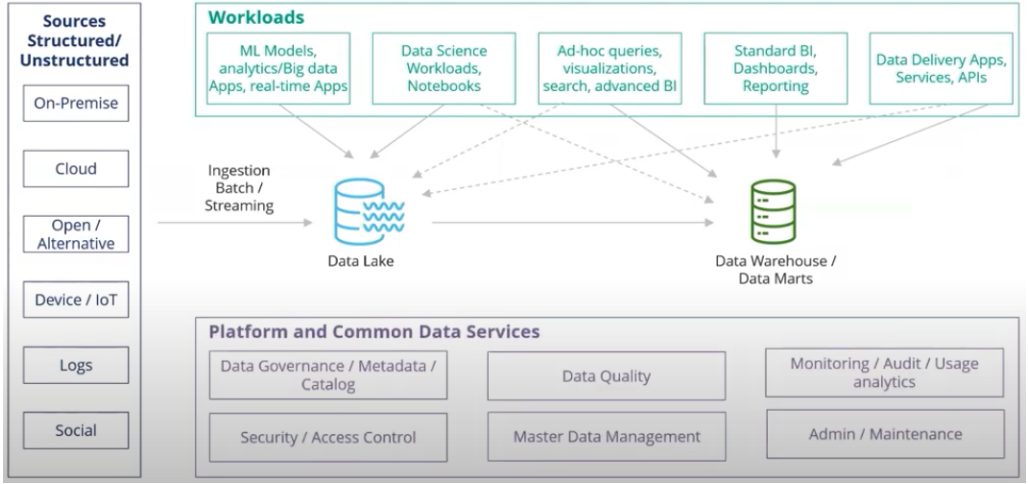
More detailed requirements

	Traditional Data Systems	Insight-driven System Needs
Data Sources	Structured, relational data from transaction systems, relational and operational data stores	Traditional sources + semi and unstructured sources: logs, web sites, social media, alternative data providers
Data Movement (Ingestion)	Amount of data that could be moved is limited	Virtually unlimited amount of data that could be moved into the system in original form at a required latency
Storage	Limited volume of stored data	Unlimited volume of data. Source of truth for all source data
Data Structure	Schema is designed upfront, before data is captured. Data format dictated by storage/technology	Schema is not fixed when data is captured. Variety of supported open formats for analytics pipelines (relational document, graph, etc.)
Data Transformations	Upfront, time consuming data cleansing, enrichment, integration and transformation, "single source of truth " of trusted and widely usable data	Add transformation for an ad-hoc data querying and data science related feature engineering

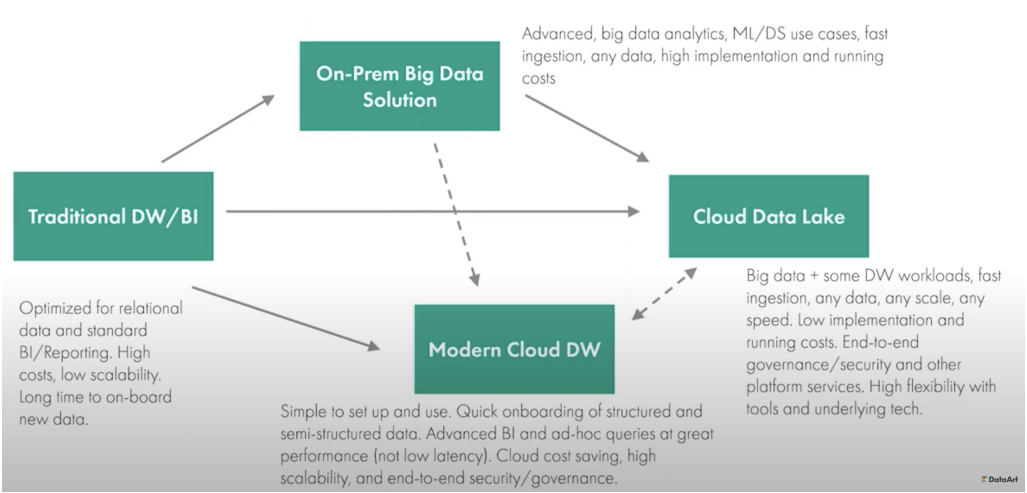
More detailed requirements (cont.)

	Traditional Data Systems	Insight-driven System Needs
Analytics	SQL Queries, BI tools, full text search	+ self-service BI, big data analytics, real-time analytics, machine learning, data exploration/visualization. Allow users to securely explore and query raw data. Easily introduce new types of analytics
Price/Performance	Highest cost storage/fastest query results	Low cost storage + performance scale/speed/cost tradeoffs
Users	Business (analysts)	+ Data Scientists, Data analysts and engineers. Developers
Data Quality	High	Low and high, depending on use case. Data quality must be transparent
Data Sharing and Collaboration	Very limited, mainly via central DW/BI team	Rich. Raw and tranformed data sets, analytical models, dashboards can be easily and securely shared

Modern Data Architecture



Data Architectures evolution and comparison - I



Data Architectures evolution and comparison - II

	Traditional DW/BI	Modern Cloud DW	On-prem Big data (e.g. Hadoop Cluster)	Cloud Data Lake
Data formats	Structured	Structured + Semi-structured	Any	Any
Schema	On-write	On-write + on-read for semi-structured	On-read	On-read
Independently scale Storage and Compute, Elasticity	No	Yes	No	Yes
Set-up and Maintenance cost	Very high	Low to Very Low	High to Very High	Low
Relational data support	Strong	Strong	Weak	Weak to Ok
New Data Ingestion	Slow to introduce	Fast if data format is supported	Fast	Fast
Data Quality	Very High	Very High + Any	Any	Any
Workloads	BI, Dashboards, Reporting	Advanced BI, Dashboards, Reporting, Ad-hoc queries	Ad-hoc queries, Data science and Machine Learning, Big Data Apps, Weak BI/Visualization	Ad-hoc queries, Data science and Machine Learning, Big Data Apps, Weak BI/Visualization

Data Architectures evolution and comparison - III

	Traditional DW/BI	Modern Cloud DW	On-prem Big data (e.g. Hadoop Cluster)	Cloud Data Lake
Data granularity	Often aggregated only	Aggregated + detail, and raw	Raw and calculated	Raw and calculated
Durability and Resilience	Limited, Expensive	Strong, Multi-cloud	Limited, Expensive	Strong, Can be multi-cloud
SLA for main workloads	Tight, optimized	Tight, optimized, elastic	Loose, unless specifically optimized	Loose, unless specifically optimized
Flexibility of architecture and tools	Low	Average	Average	High
Relational data support	Strong	Strong	Weak	Weak to Ok

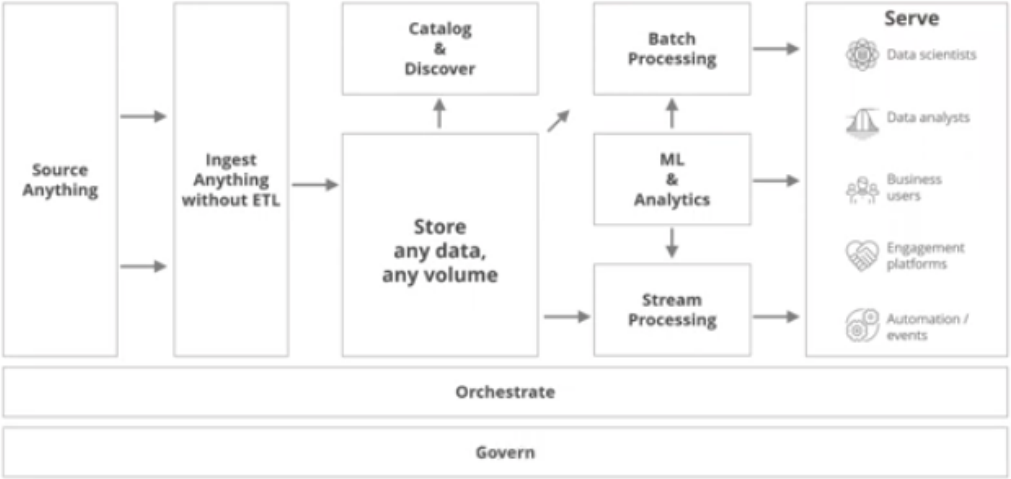
Data Lake User needs are different

Business Users – use pre-configured dashboards and reports

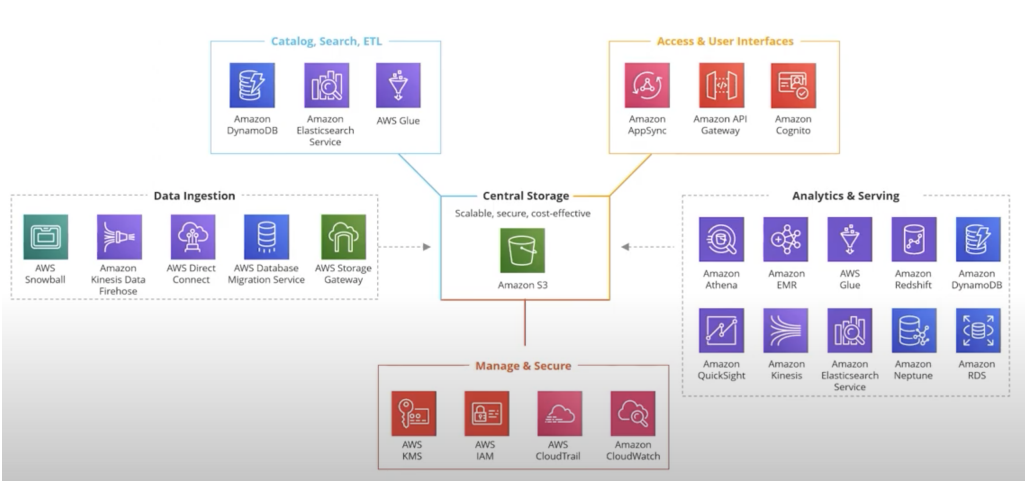
Business Analysts – use self-service BI and ad-hoc analytics, build own models to provide business insights

Data Scientists, Engineers, App Developers – perform statistical analysis and ML training, implement big data analytics to identify trends, solve business problems, optimize performance

Data Lake Components

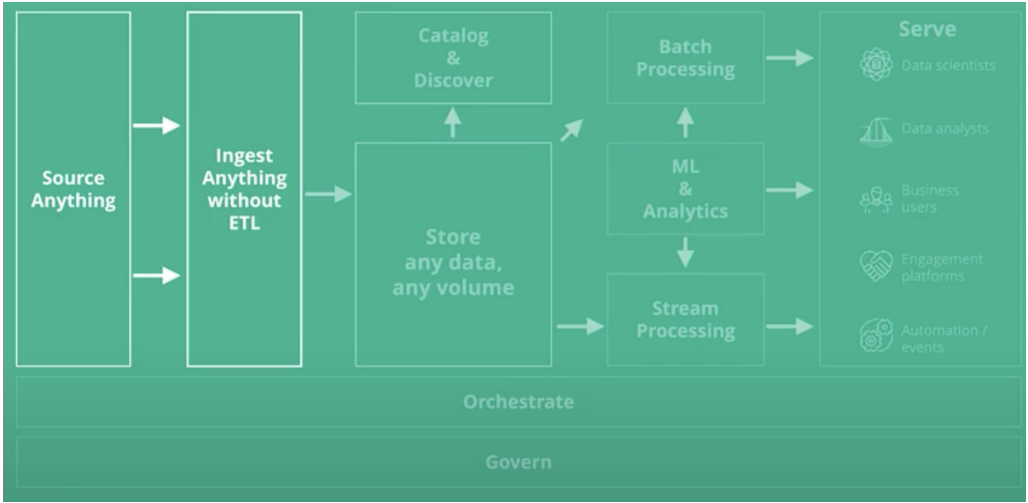


AWS Architecture I

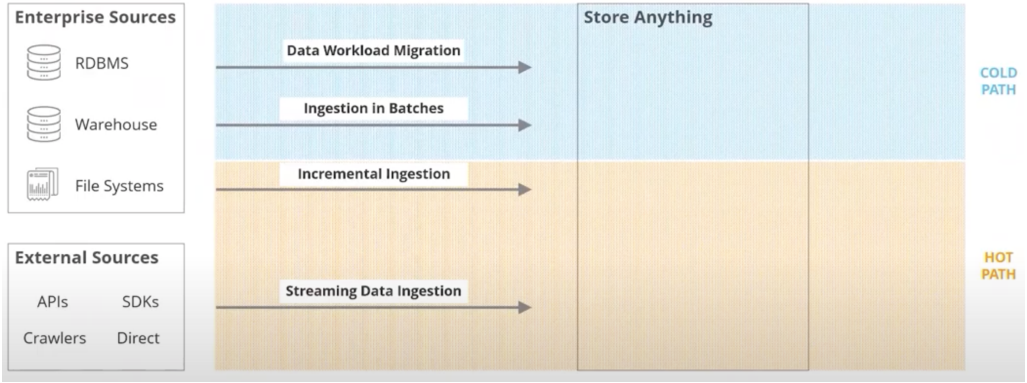


AWS Architecture II

Data Ingestion I



Data Ingestion II

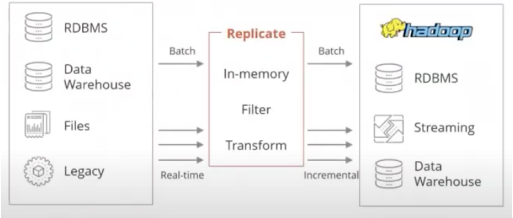


Data Ingestion options

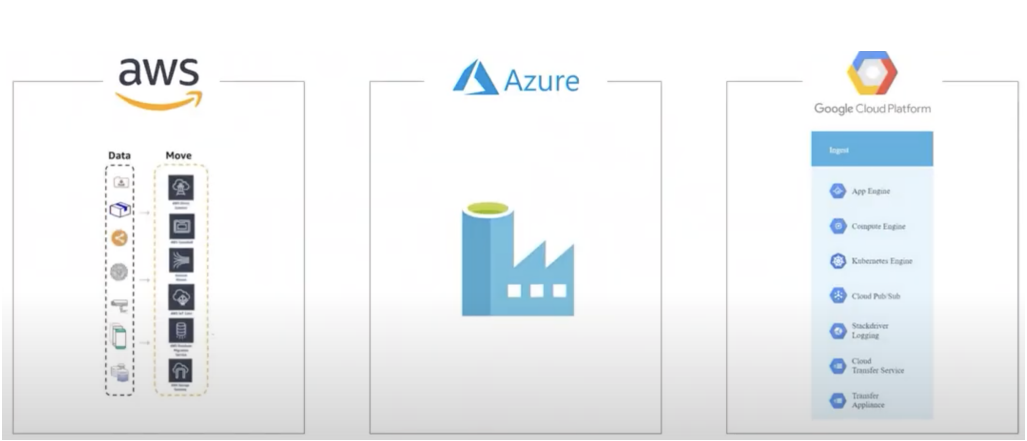
	Data Workload Migration	Incremental Ingestion	Streaming Data Ingestion
Relational Data Sources	• DB Migration Services • Transfer Services • CDC	• CDC • Streaming	
File Sources	• Storage Gateways • Snowballs/Mobiles	• Streaming/Event Sourcing • Storage Gateways	
APIs	• ISV Connectors • SDKs	• ISV Connectors • SDKs	• ISV Connectors • SDKs
Stream Sources			• Streaming Technologies

Streaming and Change Data Capture (CDC)

- Needed if
  - ingest data logs
  - ingest application events
- CDC needed if
  - continuous ingestion in Data Lake
  - capturing streaming data changes
  - database migration to cloud



No–code tools



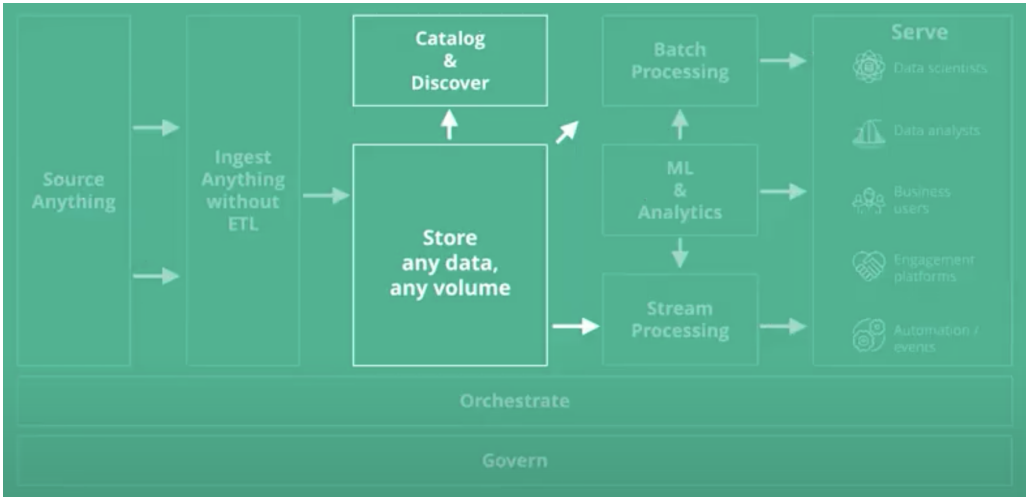
Tools which operate between the source data and the cloud data lake

- Attunity
- Matillion
- Oracle Golden Gate
- Fivetran
- ...

Data Ingestion best practices

- identify business case
- identify the right method of ingestion (the simpler the better)
- consider streaming and CDC ingestion benefits
- focus on near-term needs
  - do not over–architecture the solution, simpler data lake and ingestion tools can be set up in minutes to hours
- compress data before sending
- encrypt Personally Identifiable Information (PII)
- reduce number of files
- ensure exact processing
- automate ingestion

Storage

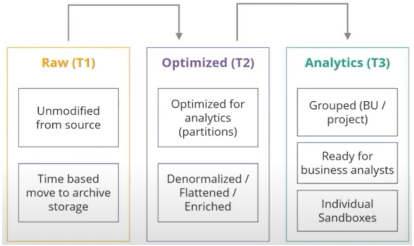

Source Anything
Ingest Anything without ETL
Catalog & Discover
Store any data, any volume
Batch Processing
ML & Analytics
Stream Processing
Serve
Data scientists
Data analysts
Business users
Engagement platforms
Automation / events
Orchestrate
Govern

Zones

Raw immutable store that cannot/should not be changed after it has been written; self–descriptive with metadata; useful for *disaster recovery*

Optimized as raw data grow, querying directly them become slower to gain speed data can be transformed in optimized formats

Analytics BI–ready and machine–learning ready data and tables e.g. after feature engineering)



Open source File Formats

File format	*Properties*
ORC	Columnar, schema in footer
Parquet	Columnar, schema in footer
AVRO	Row-major, schema and data separate
CSV	Human-readable, fixed schema
JSON	Human-readable, fixed schema

Focus: columnar storages

- keep homogenous data in a single block
- can apply strategies to compress the data in a block
- over a huge number of columns and rows reduce fragmentation, compared to row wise
- mining algorithms frequently consider entire columns, rather than entire rows
- insertions are slower, but for the intended use they represent a small part of the workload
- OLTP systems are by nature oriented to transactions, therefore row-wise format is best suited

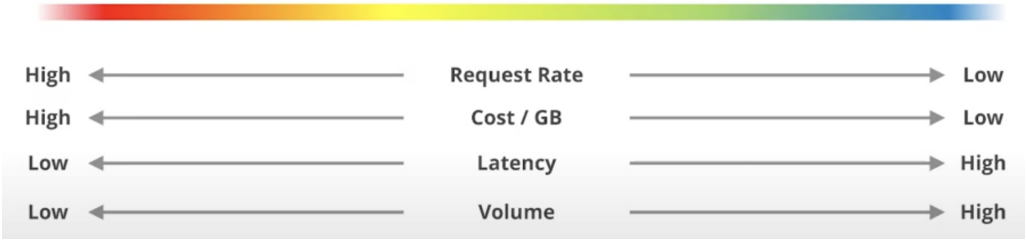
Data Catalog

- lack of schema or descriptive metadata makes hard to consume or query the data
- lack of semantic consistency makes challenging to perform analysis/mining
- without catalog the data lake risks to become a dumping area where no useful analysis is possibile

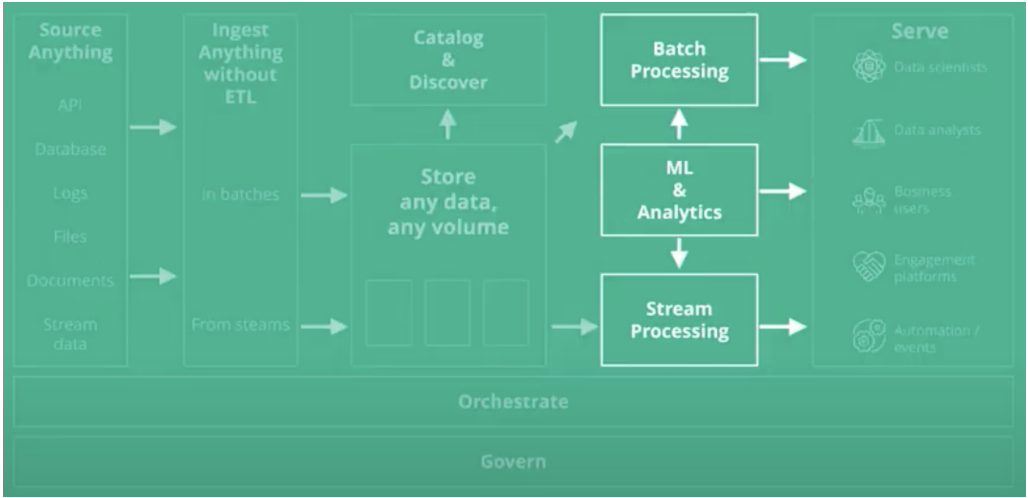
Cost Optimization

Hot Storage

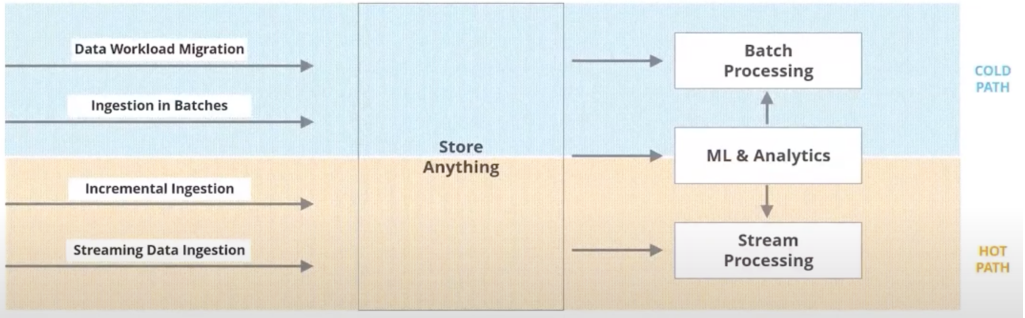
Cold Storage



Data Processing and Analytics



Processing and Analytics Needs



Interactive Analytics

- Typical users
  - Business
  - Business Analysts
  - Data Scientists
  - Developers
- Interactive queries to large data volumes
- Save query results back to data lake store

Big Data Analytics

- Typical users
  - Data Scientists
  - Developers
- Interactive queries to large data volumes
- Data Aggregations
- Data Transformations
- Complex Data Analytics

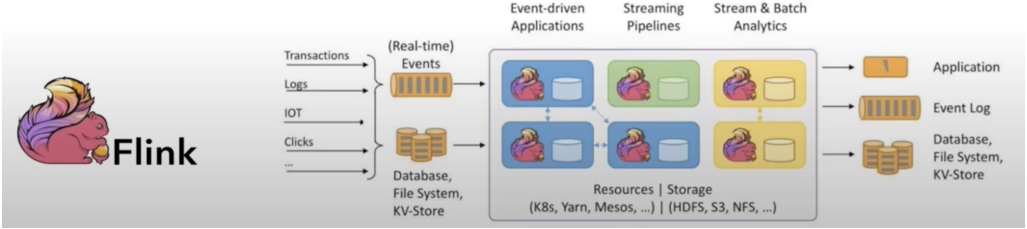
Amazon EMR

Azure HDinsight

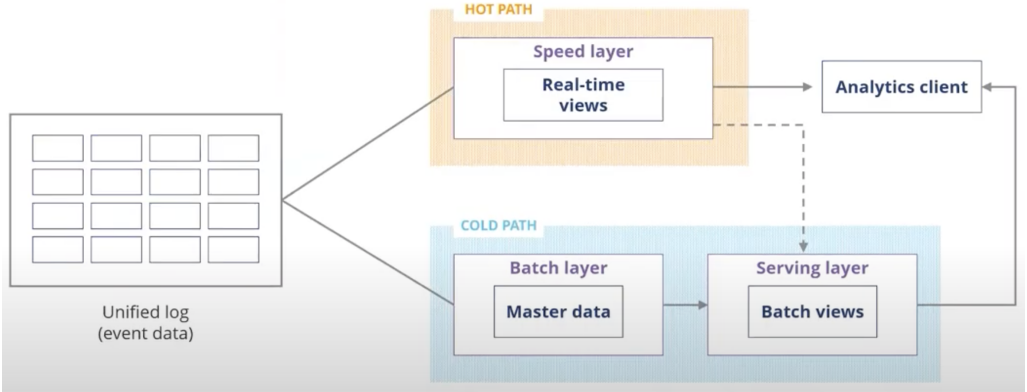
databricks

Real–Time Analytics

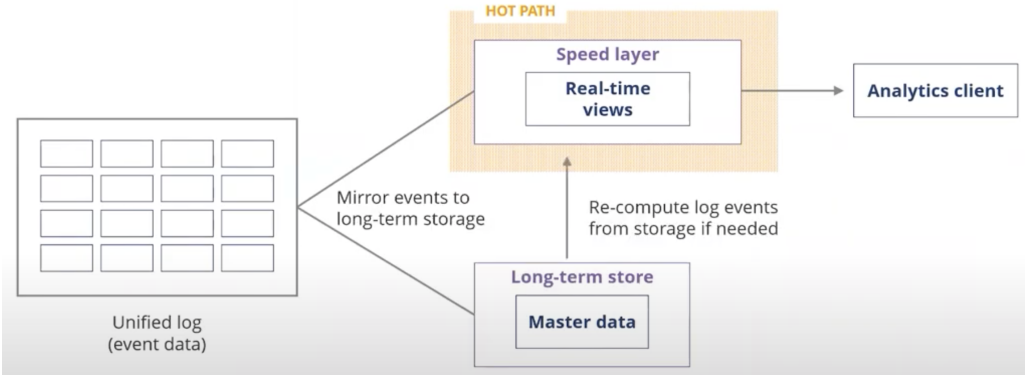
- Typical users
  - Data Scientists
  - Developers
- Streaming Analytics
- Event Detection



Lambda Lake

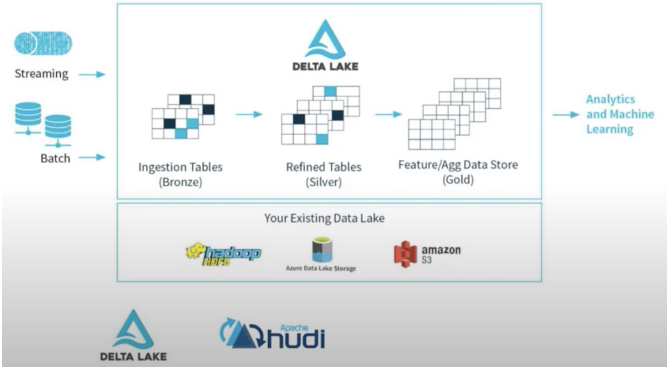

HOT PATH
Speed layer
Real-time views
Analytics client
COLD PATH
Batch layer
Master data
Serving layer
Batch views
Unified log
(event data)

Kappa Lake


HOT PATH
Speed layer
Real-time views
Analytics client
Mirror events to long-term storage
Re-compute log events from storage if needed
Long-term store
Master data
Unified log (event data)

Delta Lake: unified batch and streaming

- ACID transaction
- Scalable Metadata Handling
- Time Travel (Data Versioning)
- Unified Batch and Streaming Source and Sink
- Schema Enforcement
- Schema Evolution
- Updates, Deletes, Inserts, Upserts (= insert, on conflict update)



Metadata in a Data Lake

- Helps data engineers, data scientists, and analysts discover, understand, and use the data effectively
- Ensures that the data lake remains a valuable resource for the organization
- Typically organized and managed using metadata catalogs or data governance tools

Metadata example I

Suppose you have a data lake that stores a variety of data, including customer information, sales transactions, and product inventory

- **Data Source** Information about where the data originated, such as the source system, data provider, or data feed. For example
  - *Data Source Name* CRM System
  - *Data Source Location* S3 bucket (if stored in AWS)
- **Data Schema** Describes the structure of the data, including the names and data types of columns or attributes. For instance:
  - *Schema Name* CustomerDataSchema
  - *Column Names* CustomerID, FirstName, LastName, Email, Phone, Address, etc.

Metadata example II

Suppose you have a data lake that stores a variety of data, including customer information, sales transactions, and product inventory

- **Data Format** Specifies the file format or encoding used for the data files, which can be important for data processing and analytics. For example
  - *Format* Parquet
  - *Compression* Snappy
- **Data Quality Metrics** Information about data quality, such as the percentage of missing values or data accuracy statistics.

Metadata example III

Suppose you have a data lake that stores a variety of data, including customer information, sales transactions, and product inventory

- **Data Lifecycle** Information about how long the data should be retained in the data lake, data retention policies, and archiving rules.
- **Data Ownership** Information about the team or department responsible for managing and maintaining the data.
- **Data Lineage** Tracks the data's lineage, including its transformation processes, dependencies, and any data transformations applied to it.

Metadata example IV

Suppose you have a data lake that stores a variety of data, including customer information, sales transactions, and product inventory

- **Access Control** Metadata can specify who has access to the data, what permissions they have, and any security measures in place.
- **Data Classification** Indicates the sensitivity or classification level of the data, which can be crucial for compliance with data regulations (e.g., GDPR, HIPAA).
- **Data Usage Information** Records how frequently the data is accessed, who accesses it, and what types of analytics or applications use the data.

Final thoughts

Final thoughts

- Use Data Lake as a landing zone for all your data

Final thoughts

- Use Data Lake as a landing zone for all your data
- Secure data with Role–based access control

Final thoughts

- Use Data Lake as a landing zone for all your data
- Secure data with Role–based access control
- Avoid *Data Swamps*

Final thoughts

- Use Data Lake as a landing zone for all your data
- Secure data with Role–based access control
- Avoid *Data Swamps*
  - catalogs help

Final thoughts

- Use Data Lake as a landing zone for all your data
- Secure data with Role–based access control
- Avoid *Data Swamps*
  - catalogs help
- Insight–driven companies use Data Lakes

Final thoughts

- Use Data Lake as a landing zone for all your data
- Secure data with Role–based access control
- Avoid *Data Swamps*
  - catalogs help
- Insight–driven companies use Data Lakes
  - 80% of the *data payload* is unstructured data

Final thoughts

- Use Data Lake as a landing zone for all your data
- Secure data with Role–based access control
- Avoid *Data Swamps*
  - catalogs help
- Insight–driven companies use Data Lakes
  - 80% of the *data payload* is unstructured data
- Data Lakes can be enabled in a matter of days

Main Data Warehouse software vendors

Rank	*Technology*	*Description*	*Companies*	*Share*
1	Snowflake	The only DW built for the cloud. Snowflake delivers the performance, concurrency and simplicity needed to store and analyze all of an organization's data in one solution. Snowflake combines the power of data warehousing, the flexibility of big data platforms and the elasticity of the cloud at a fraction of the cost of traditional solutions	7.831	28.38%
2	Apache Hive	Organizes large datasets. It provides tools to access data using SQL easily, a machine to assess structure on a range of data formats, query execution, sub-second query retrieval, and procedural language	3.021	10.95%
3	SAP Business Warehouse	It is SAP's Enterprise DW product. It can transform and consolidate business information from virtually any source system	2.933	10.63%
4	Google BigQuery	Cloud-based big data analytics web service for processing very large read-only data sets. BigQuery was designed for analyzing data on the order of billions of rows, using a SQL-like syntax. It runs on the Google Cloud Storage infrastructure and can be accessed with a REST-oriented application program interface (API).	2.732	9.90%
5	Amazon Redshift	Internet hosting service and DW product which forms part of the larger cloud-computing platform Amazon Web Services. It allows to run complex analytic queries against petabytes of structured data, using sophisticated query optimization, columnar storage on high-performance local disks, and massively parallel query execution	2.426	8.79%
6	Oracle Autonomous DW Cloud	Oracle Database specifically configured and optimized to handle the size of data and types of queries that are intrinsic to data warehousing	943	3.42%
7	IBM Netezza DataWarehouse Appliances	An IBM Netezza appliance consists of a high-performance hardware platform and optimized database query engine software that work together	937	3.40%
8	Fivetran	Fully-managed data pipeline.	898	3.25%

Microsoft Azure I

Compute Virtual machines (VMs), containers, serverless computing with Azure Functions, and Azure Kubernetes Service (AKS) for container orchestration.

Storage Scalable and highly available storage solutions, including blob storage, file storage, table storage, and disk storage. Azure also offers services like Azure Data Lake Storage and Azure SQL Database.

Networking Networking services such as Virtual Network (VNet) for creating isolated network environments, Azure Load Balancer for load distribution, Azure VPN Gateway for secure connectivity, and Azure Application Gateway for web traffic management.

Microsoft Azure II

Databases Range of database services, including Azure SQL Database, Azure Cosmos DB (NoSQL database), Azure Database for MySQL, Azure Database for PostgreSQL, and more.

Analytics and Big Data Services like Azure Synapse Analytics (formerly SQL Data Warehouse), Azure HDInsight, and Azure Data Factory enable organizations to process and analyze large datasets.

AI and Machine Learning Tools and services for AI and machine learning, such as Azure Machine Learning, Azure Cognitive Services, and Azure Databricks.

Internet of Things (IoT) Azure IoT Hub and Azure IoT Central help organizations connect, monitor, and manage IoT devices and data.

Microsoft Azure III

DevOps and Application Development Azure DevOps services, Azure App Service, and Azure Kubernetes Service (AKS) facilitate application development, deployment, and DevOps practices.

Security and Identity Various security and identity solutions, including Azure Active Directory (Azure AD), Azure Key Vault, and Azure Security Center.

Integration and Messaging Azure Logic Apps and Azure Service Bus enable organizations to create workflows and integrate applications and services.

Management and Governance Tools for managing and governing resources, including Azure Monitor, Azure Policy, and Azure Resource Manager.

Microsoft Azure IV

Hybrid Solutions Hybrid cloud solutions, allowing organizations to seamlessly integrate on-premises infrastructure with Azure resources.

IoT Edge Azure IoT Edge extends Azure IoT capabilities to the edge, enabling processing and analysis of data closer to IoT devices.